

# Relatório do EP3

## MAC0352 – Redes de Computadores e Sistemas Distribuídos – 2/2019

*LuizGiroto* (8941189), *VictorAraujo* (9793756)

17/10/2019

### 1 Passo 0

Na definição do protocolo *OpenFlow*, o que um switch faz toda vez que ele recebe um pacote que ele nunca recebeu antes?

Ao receber um pacote que ele nunca recebeu antes, um switch *OpenFlow* encaminha este pacote para o controlador *OpenFlow*. Este então determina a forma com que será lidado este pacote.

### 2 Passo 2

Com o acesso à Internet funcionando em sua rede local, instale na VM o programa `traceroute` usando `sudo apt install traceroute` e escreva abaixo a saída do comando `sudo traceroute -I www.inria.fr`. Pela saída do comando, a partir de qual salto os pacotes alcançaram um roteador na Europa? Como você chegou a essa conclusão?

```
<traceroute to www.inria.fr (128.93.162.84), 30 hops max, 60 byte packets
 1  10.0.2.2 (10.0.2.2)  0.110 ms  0.095 ms  0.167 ms
 2  192.168.15.1 (192.168.15.1)  1.193 ms  1.457 ms  1.367 ms
 3  * * *
 4  187-100-163-72.dsl.telesp.net.br (187.100.163.72)  50.045 ms  49.909 ms  49$
 5  152-255-158-36.user.vivozap.com.br (152.255.158.36)  7.161 ms  7.071 ms  6.$
 6  84.16.9.109 (84.16.9.109)  13.040 ms  9.830 ms  9.600 ms
 7  94.142.98.177 (94.142.98.177)  124.261 ms  120.983 ms  125.332 ms
 8  84.16.15.129 (84.16.15.129)  120.564 ms  124.784 ms  124.723 ms
 9  213.140.36.89 (213.140.36.89)  124.474 ms  124.306 ms  124.158 ms
10  ip4.gtt.net (208.116.240.149)  145.831 ms  155.251 ms  155.017 ms
11  et-3-3-0.cr4-par7.ip4.gtt.net (213.200.119.214)  230.782 ms  228.760 ms  22$
12  renater-gw-ix1.gtt.net (77.67.123.206)  244.947 ms  246.479 ms  246.284 ms
13  tel1-1-inria-rtr-021.noc.renater.fr (193.51.177.107)  240.280 ms  240.100 ms$
14  inria-rocquencourt-tel-4-inria-rtr-021.noc.renater.fr (193.51.184.177)  239$
15  unit240-reth1-vfw-ext-dc1.inria.fr (192.93.122.19)  242.478 ms  241.464 ms $
16  ezp3.inria.fr (128.93.162.84)  238.068 ms  243.845 ms  244.529 ms>
```

A partir do décimo-primeiro salto, os pacotes alcançaram um roteador na Europa. É possível perceber isto pelo tempo de resposta similar ao pacote encontrado no décimo-terceiro salto, o qual possui identificador de endereços da França (fr).

### 3 Passo 3 - Parte 1

Execute o comando `iperf`, conforme descrito no tutorial, antes de usar a opção `--switch user`, 5 vezes. Escreva abaixo o valor médio e o intervalo de confiança da taxa retornada (considere sempre o primeiro valor do vetor retornado).

```
1:
*** Iperf: testing TCP bandwidth between h1 and h3
*** Results: ['16.7 Gbits/sec', '16.8 Gbits/sec']
```

```
2:
*** Iperf: testing TCP bandwidth between h1 and h3
*** Results: ['11.0 Gbits/sec', '11.1 Gbits/sec']
```

```
3:
*** Iperf: testing TCP bandwidth between h1 and h3
*** Results: ['12.5 Gbits/sec', '12.5 Gbits/sec']
```

```
4:
*** Iperf: testing TCP bandwidth between h1 and h3
*** Results: ['16.8 Gbits/sec', '16.8 Gbits/sec']
```

```
5:
*** Iperf: testing TCP bandwidth between h1 and h3
*** Results: ['17.2 Gbits/sec', '17.2 Gbits/sec']
```

Média: 14.84 Gbits/sec. Intervalo de confiança: 95% CI [12.6, 17.1] Gbits/sec

### 4 Passo 3 - Parte 2

Execute o comando `iperf`, conforme descrito no tutorial, com a opção `--switch user`, 5 vezes. Escreva abaixo o valor médio e o intervalo de confiança da taxa retornada (considere sempre o primeiro valor do vetor retornado). O resultado agora corresponde a quantas vezes menos o da Seção anterior? Qual o motivo dessa diferença?

```
1:
*** Iperf: testing TCP bandwidth between h1 and h3
*** Results: ['274 Mbits/sec', '275 Mbits/sec']
```

```
2:
*** Iperf: testing TCP bandwidth between h1 and h3
*** Results: ['274 Mbits/sec', '275 Mbits/sec']
```

```
3:
*** Iperf: testing TCP bandwidth between h1 and h3
*** Results: ['275 Mbits/sec', '277 Mbits/sec']

4:
*** Iperf: testing TCP bandwidth between h1 and h3
*** Results: ['272 Mbits/sec', '273 Mbits/sec']

5:
*** Iperf: testing TCP bandwidth between h1 and h3
*** Results: ['275 Mbits/sec', '277 Mbits/sec']
```

Média: 274 Mbits/sec. Intervalo de confiança: 95% CI [273, 275] Mbits/sec

A largura de banda é aproximadamente 54 vezes menor do que nos testes da seção anterior. Isto ocorre pois agora, ao invés dos pacotes serem trocados apenas por via do *kernel*, elas devem iniciar seu caminho *user-space*, passando então para o *kernel*, e novamente para o *user-space*, o que aumenta bastante o tempo de transmissão dos pacotes.

## 5 Passo 4 - Parte 1

Execute o comando `iperf`, conforme descrito no tutorial, usando o controlador `of_tutorial.py` original sem modificação, 5 vezes. Escreva abaixo o valor médio e o intervalo de confiança da taxa retornada (considere sempre o primeiro valor do vetor retornado). O resultado agora corresponde a quantas vezes menos o da Seção 3? Qual o motivo para essa diferença? Use a saída do comando `tcpdump`, deixando claro em quais computadores virtuais ele foi executado, para justificar a sua resposta.

## 6 Passo 4 - Parte 2

Execute o comando `iperf`, conforme descrito no tutorial, usando o seu controlador `switch.py`, 5 vezes. Escreva abaixo o valor médio e o intervalo de confiança da taxa retornada (considere sempre o primeiro valor do vetor retornado). O resultado agora corresponde a quantas vezes mais o da Seção anterior? Qual o motivo dessa diferença? Use a saída do comando `tcpdump`, deixando claro em quais computadores virtuais ele foi executado, para justificar a sua resposta.

## 7 Passo 4 - Parte 3

Execute o comando `iperf`, conforme descrito no tutorial, usando o seu controlador `switch.py` melhorado, 5 vezes. Escreva abaixo o valor médio e o intervalo de confiança da taxa retornada (considere sempre o primeiro valor do vetor retornado). O resultado agora corresponde a quantas vezes mais o da Seção anterior? Qual o motivo dessa diferença? Use a saída do comando `tcpdump`, deixando claro em quais computadores virtuais ele foi executado, e saídas do comando `sudo ovs-ofctl`, com os devidos parâmetros, para justificar a sua resposta.

## 8 Passo 5

Explique a lógica implementada no seu controlador `firewall.py` e mostre saídas de comandos que comprovem que ele está de fato funcionando (saídas dos comandos `tcpdump`, `sudo ovs-ofctl, nc`, `iperf` e `telnet` são recomendadas)

## 9 Configuração dos computadores virtual e real usados nas medições (se foi usado mais de um, especifique qual passo foi feito com cada um)

## 10 Referências

- 
-